IN THE CLAIMS:

The text of all pending claims, (including withdrawn claims) is set forth below. Cancelled and not entered claims are indicated with claim number and status only. The claims as listed below show added text with underlining and deleted text with ~~strikethrough~~. The status of each claim is indicated with one of (original), (currently amended), (cancelled), (withdrawn), (new), (previously presented), or (not entered).

Claims 1-5 (cancelled).

6. (CURRENTLY AMENDED) A method for decoding a data sequence of K information bits encoded with aid of a binary convolutional code, comprising:

~~precisely~~ calculating in a first operation, on a trellis diagram having trellis segments, first ~~metrics~~ metric values of all trellis segments for a forward direction and for a backward direction using a MaxLogMAP algorithm without a decoding window;

selecting a first set of trellis segments as first interpolation nodes from the first operation;

storing first associated ~~metrics~~ metric values for the first set of trellis segments in a first memory level;

iteratively performing a number of n operations until the metric values for the forward direction and for the backward direction meet in one trellis segment, including:

calculating in an i-th operation for $1< i \leq n$, i-th ~~metrics~~ metric values of internodal trellis segments positioned between the interpolation nodes of an i-1-th operation for ~~both~~ the forward and the backward directions using stored ~~metrics~~ metric values of interpolation nodes of the i-1-th operation;~~,~~

selecting an i-th set of trellis segments as i-th interpolation nodes from the i-th operation;~~,~~ and

storing i-th associated ~~metrics~~ metric values for the i-th set of trellis segments in an i-th memory level;

~~repeating said calculating, selecting and storing of the i-th operation metrics values n -1 times until the metrics values of the forward and backward directions meet in one trellis segment;~~ and

carrying out a decision process to calculate soft output values for decoding.

7. (CURRENTLY AMENDED) A method according to claim 6, further comprising:

assigning a memory depth of $\delta 1$ for each direction of the first memory level, and

wherein said storing of the first associated ~~metrics~~ metric values in the first memory level uses each K/δ1-th trellis segment.

8. (CURRENTLY AMENDED) A method according to claim 7, further comprising:

assigning a memory depth of δi for each direction of the i-th memory level, and

wherein said storing of the first associated ~~metrics~~ metric values in the first memory level uses each K/δ1/.../δi-th trellis segment.

9. (PREVIOUSLY PRESENTED) A method according to claim 8, wherein a delayed decision phase is used in calculating the soft output values for terminated codes.

10. (CURRENTLY AMENDED) A method according to claim 9, wherein the decoding is carried out on ~~precisely one~~ a single application-specific module.

11. (CURRENTLY AMENDED) A method according to claim 8, wherein the decoding is carried out on ~~precisely one~~ a single application-specific module.

12. (PREVIOUSLY PRESENTED) A method according to claim 7, wherein a delayed decision phase is used in calculating the soft output values for terminated codes.

13. (CURRENTLY AMENDED) A method according to claim 7, wherein the decoding is carried out on ~~precisely one~~ a single application-specific module.

14. (CURRENTLY AMENDED) A method according to claim 6, further comprising:

assigning a memory depth of δi for each direction of the i-th memory level, and

wherein said storing of the first associated ~~metrics~~ metric values in the first memory level uses each K/δ1/.../δi-th trellis segment.

15. (PREVIOUSLY PRESENTED) A method according to claim 6, wherein a delayed decision phase is used in calculating the soft output values for terminated codes.

16. (CURRENTLY AMENDED) A method according to claim 6, wherein the decoding is carried out on ~~precisely one~~ a single application-specific module.